Applications of GAs and QGAs on theoretical fully autonomous road networks

University of Birmingham

Sam Barrett
sjb786@student.bham.ac.uk

November 4, 2020

Contents

Abstract

Chapter 1

Introduction

Chapter 2

Background

2.1 Genetic Algorithms

2.1.1 History

Genetic algorithms are optimisation techniques that employ the same rationale as classical Evolution as seen in nature.

Genetic Algorithms can trace their origins back to the late 1960s when they were first proposed by John Holland. Holland went on to write the first book on the subject titled *Adaptation in Natural and Artificial Systems* [1] in 1975. The field did not find much reception with Holland stating in the preface to the 1992 rerun:

> *"When this book was originally published I was very optimistic, envisioning extensive reviews and a kind of 'best seller' in the realm of monographs. Alas! That did not happen."*

However, in the early nineties, Genetic algorithms surged in popularity along with Artificial Intelligence as a whole leading to Holland republishing his book and solidifying his position as the field's founder.

2.1.2 Definition

In a general sense, optimisation techniques work to find the set of parameters $\mathcal{P}$ that minimise an objective function $\mathcal{F}$. Genetic algorithms approach this by representing these sets as individuals in a population, P. Over the course of multiple generations, the best solutions are determined and promoted until termination criteria are met or the maximum number of generations is reached.

As our candidates are essentially a collection of parameters to the function we are trying to optimise, we can extend our metaphor further by mapping each element of a individual to a *gene* in a individual's genome.

The representation we use in a GA is problem specific. Often we have to provide functions to facilitate the mapping between the problem specific set of possible solutions and the encoded genotype space in which we optimise. The most basic representation being a string of binary numbers.

Genetic algorithms are both *probabilistically optimal* and *probabilistically complete*[2] meaning that: given infinite time, not only will the algorithm find *a* solution, (if one exists), it will find **the** optimal solution from the set of all possible solutions, $\mathcal{P}^*$.

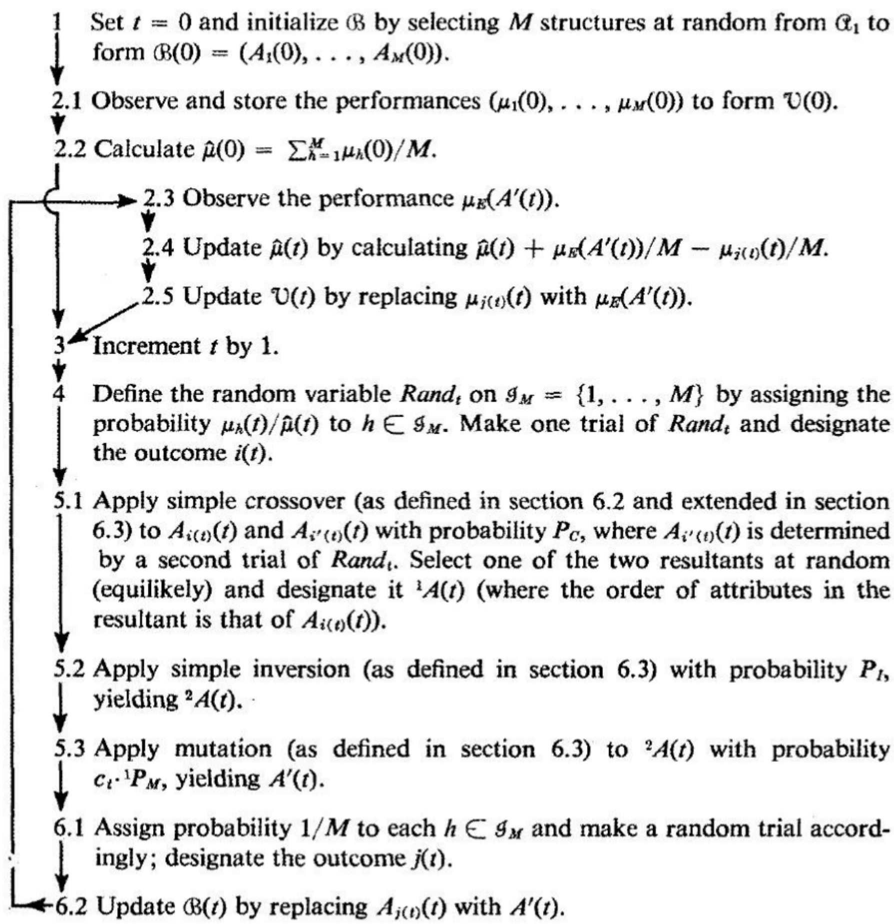
Algorithm 1: Modern Generic Genetic Algorithm

Result: Best Solution, $p_{\texttt{best}}$
Generate initial population, P_0 of size n;
Evaluate fitness of each individual in P_0, $\{F(p_{0,1}, \ldots, p_{0,n})\}$;
while *termination criteria are not met* **do**
    Selection: Select individuals from P_t based on their fitness;
    Variation: Apply variation operators to parents from P_t to produce offspring;
    Evaluation: Evaluate the fitness of the newly bred individuals;
    Reproduction: Generate a new population P_{t+1} using individuals from P_t as well as the newly bred candidates.;
    t++
end
return $p_{\texttt{best}}$

As you can see from Figure 2.1 and Algorithm 1 the overall shape of GAs has not changed substantially over the course of the past 50 years. Being comprised of a series of operations that a starting population is piped through until criteria are met.

2.1.3 Selection

The selection procedure



1 Set $t = 0$ and initialize $\mathcal{B}$ by selecting M structures at random from $\mathcal{A}_1$ to form $\mathcal{B}(0) = (A_1(0), \ldots, A_M(0))$.

2.1 Observe and store the performances $(\mu_1(0), \ldots, \mu_M(0))$ to form $\mathcal{U}(0)$.

2.2 Calculate $\hat{\mu}(0) = \sum_{h=1}^{M} \mu_h(0)/M$.

2.3 Observe the performance $\mu_E(A'(t))$.

2.4 Update $\hat{\mu}(t)$ by calculating $\hat{\mu}(t) + \mu_E(A'(t))/M - \mu_{j(t)}(t)/M$.

2.5 Update $\mathcal{U}(t)$ by replacing $\mu_{j(t)}(t)$ with $\mu_E(A'(t))$.

3 Increment t by 1.

4 Define the random variable $Rand_t$ on $\mathcal{J}_M = \{1, \ldots, M\}$ by assigning the probability $\mu_h(t)/\hat{\mu}(t)$ to $h \in \mathcal{J}_M$. Make one trial of $Rand_t$ and designate the outcome $i(t)$.

5.1 Apply simple crossover (as defined in section 6.2 and extended in section 6.3) to $A_{i(t)}(t)$ and $A_{i'(t)}(t)$ with probability P_C, where $A_{i'(t)}(t)$ is determined by a second trial of $Rand_t$. Select one of the two resultants at random (equilikely) and designate it ${}^1A(t)$ (where the order of attributes in the resultant is that of $A_{i(t)}(t)$).

5.2 Apply simple inversion (as defined in section 6.3) with probability P_I, yielding ${}^2A(t)$.

5.3 Apply mutation (as defined in section 6.3) to ${}^2A(t)$ with probability $c_t \cdot {}^1P_M$, yielding $A'(t)$.

6.1 Assign probability $1/M$ to each $h \in \mathcal{J}_M$ and make a random trial accordingly; designate the outcome $j(t)$.

6.2 Update $\mathcal{B}(t)$ by replacing $A_{j(t)}(t)$ with $A'(t)$.

Figure 2.1: GA algorithm outlined in Holland's Original Book[1]

2.1.4 Variation

2.1.5 Evaluation

2.1.6 Reproduction

2.1.7 Examples

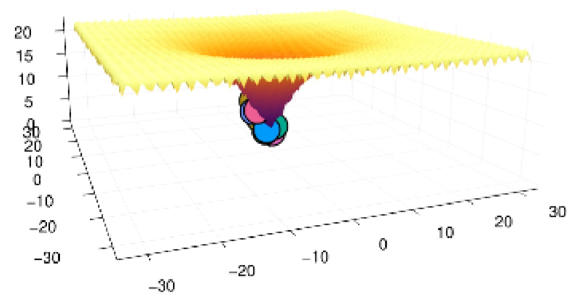


Figure 2.2: A basic GA applied to a search space defined using the Ackley function [3]

2.2 Autonomous Road Networks

2.3 Quantum Genetic Algorithms

2.4 Alternative Technologies

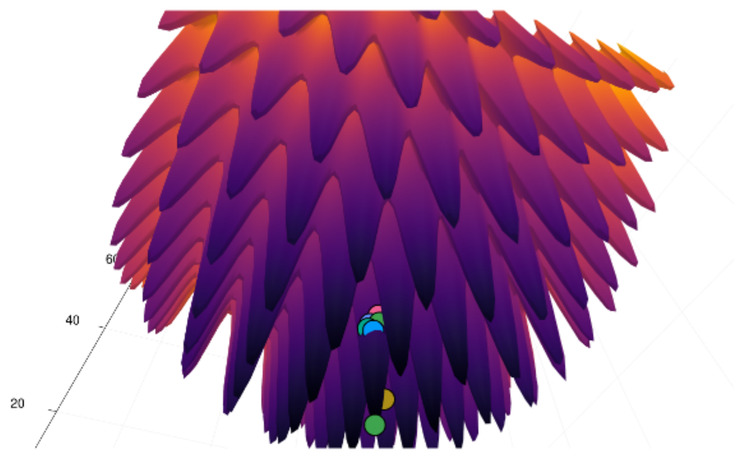


Figure 2.3: A basic GA applied to a search space defined using the Rastrigin function [4]

Chapter 3

Literature Review

3.1 Classical GAs

3.2 Quantum GAs

3.2.1 Quantum Computing

Chapter 4

Classical Approach

4.1 Approach

4.2 Implementation

4.2.1 Language Choice

4.3 Results

Chapter 5

Quantum Approach

5.1 Approach

5.2 Implementation

5.3 Results

Chapter 6

Evaluation

Chapter 7

Conclusion

Bibliography

[1] John H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press, Cambridge, UNITED STATES, 1992.

[2] Rahul Kala. *On-Road Intelligent Vehicles: Motion Planning for Intelligent Transportation Systems / Rahul Kala.* Butterworth-Heinemann is an imprint of Elsevier, Kidlington, Oxford, UK, 2016.

[3] David Ackley. *A Connectionist Machine for Genetic Hillclimbing.* Springer Science & Business Media, December 2012.

[4] L. A. Rastrigin. Systems of extremal control., 1974.